Predicción de ventas minoristas (RETAIL SALES)

Jorge Martínez López

*Tecnológico de Monterrey campus Querétaro*

*Querétaro, México*

A01704518@tec.mx

jorgemartinez2555@hotmail.com

***Abstract*— Este documento es un ejemplo de formato apegado a las normas de IEEE para escribir artículos representativos de un proyecto realizado. Los autores deben seguir las instrucciones, incluyendo formato y tamaño de papel para mantener el estándar de publicación. Este documento puede interpretarse como un set de instrucciones para escribir su artículo o como una plantilla para hacerlo.**

I. INTRODUCCIÓN

El surgimiento de diferentes tecnologías vanguardistas a generado un crecimiento exponencial en ámbito académico, laboral, social, entre otras, las cuales podemos destacar el uso de inteligencia artificial para resolver dudas, optimizar y tomar decisiones.

No obstante, son contados los sectores de la población que entiende que hay trasfondo en estos algoritmos que simulan el aprendizaje. Por ende, por esta brecha de conocimiento se generan ideas ambiguas de su funcionamiento, al contrario, su implementación ayuda a maximizar el flujo de trabajo como de productividad.

En la actualidad, gran parte de los sectores mercantiles pequeños y medianos, se ven solapados por aquellas empresas que empiezan implementar el uso de IA a sus procesos, ocasionando que los productores minoritas quiebren.

II. HISTORIA ML - PROBLEMA

El génesis de la inteligencia artificial surge entre 1943-1955, parte de tres fuentes de conocimiento de la época, las cuales radica en la filosofía básica y el funcionamiento de las neuronas, el análisis de la Lógica proporcional de Russell y la Teoría Computacional de Turing, estos trabajos fueron los cimientos de lo que se conoce hoy como redes neuronales. Sin embargo, hasta el 2010 fue que empezó a tener avances significativos las redes neuronales gracias a su desempeño en competencias, por otro lado, antes de que esto sucediera. A mediados de los 80°, dentro del área de AI surgieron algoritmos que simulan agentes inteligentes, que prácticamente son algoritmos que maximizar sus métricas de rendimiento para alcanzar un objetivo en particular, estos se conocen como autómatas, ejemplos A\*, Knapsack, BFS, etc.., dando origen a dos categorías básicas de algoritmos, supervisados y no supervisados (*Artificial Intelligence: A Modern Approach, 4th Global Ed.*, 2022).

En este trabajo profundizaremos en el área de aprendizaje supervisado, que son las bases de una red neuronal.

*A. Situación Actual*

En la actualidad México cuanta un crecimiento exponencial del sector económicas dedicadas al comercio al por menor, se estima que de 2024 a 2027 el mercado minorista crecerá un 33.3%, para llegar a los 664 mil millones de dólares. Además, según Instituto Nacional de Estadística y Geografía, 2020 esta industria representó aproximadamente el 11.5% del Producto Interno Bruto, que engloba al 17% de la población económicamente activa (Ruiz-Healy, 2024).

Sin embargo, el mantener un comercio de este tipo es todo un desafío, se enfrenta a entornos económicos y social cambiantes.

En México tres de cada cuatro tienditas se ven obligadas a cerrar, debido a los hábitos de consumo y pobreza del cliente y del entorno, una encuetas realizada por la Anpec (Alianza Nacional de Pequeños Comerciantes), los datos arrojaron que 94% de las tienditas observa que el consumo de su cliente es bajo e insuficiente y 64% ha visto que los clientes compran porciones menore al del producto (Elizabeth Meza Rodríguez, 2022).

Por ende, para poder hacer frente a las adversidades a las que se enfrentan la mayoría de las tiendas minoristas.

*Podría ser una respuesta el implementar modelos de Machine Learning para impulsar los negocios minoritas, tomando en cuenta su complejidad, su asertividad y su implementación.*

III. MODELOS

Como mencionamos anteriormente, existen dos categorías de modelos de Machine Learning, supervisados y no supervisados, cada una de ellas tiene su complejidad al momento de aplicarlas, para poder abordar la situación de la manera óptima y menos trivial, es necesario acudir aquellos modelos que demanden menos recurso computacional y de almacenamiento, y que puedan ser entrenados con un conjunto de datos más pequeños.

Por ello, trabajaremos con la categoría de entrenamiento supervisado, y los dos modelos que destacan es Regresión Lineal y Regresión Logística.

*A. Regresiones Lineales*

Una regresión lineal es el modelo más simple de predicción, pero el más poderoso si se implementa bien.

Un modelo de este tipo cuenta con una variable dependiente o conocida como Label que será la que va a predecir, luego X que es una variable independiente que es conocida como Features, y serán las variables de entrada para nuestro modelo, también tenemos la pendiente o beta que es m, que será la razón de cambio o mejor dicho describe como va a variar nuestro modelo, por último, el B que es conocido como el Bias que es la intercepción de nuestro modelo cuando es cero, o como va iniciar nuestro modelo.

Lo excelente de una regresión lineal, es que puede contener de una a N variables de entrada para hacer predicciones, esto se conoce como múltiples dimensionalidades.

*B. Regresiones Logística*

Una regresión logística tiene la finalidad de clasificar, esta surge de multiplicar la Regresión Lineal por una función de activación sigmoidal. La finalidad de esta función de activación convierte el valor real de xm+b entre un valor de 0 y 1, que se interpreta como la probabilidad de que la muestra pertenezca a una clase positiva, ejemplo si la probabilidad es mayor o igual a 0.5, se clasifica como clase positiva (1), y si es menor será de tipo negativo (0). Este modelo se le conoce como clasificación binaria.

No obstante, se puede utilizar múltiples parámetros de entrada para alimentar el modelo y hacer más complejo para clasificación predicción correcta.

IV. DATASET

Antes de empezar con la programación y optimización de nuestros modelos, es primordial escoger un banco de información con la que estaremos alimentando nuestro modelo. Al ser una problemática de predecir tendencias de ventas, escogí una Dataset, que podamos describir el suceso y así replicar el modelo para cualquier tienda minorista.

La Dataset fue obtenida de la organización data Montgomery, la tabla contiene una lista de ventas y movimiento por producto y departamento por mes.

El nombre de la Dataset es Warehouse and Retail Sales.

Sus features son:

* YEAR (Año)
* MONTH (Mes)
* SUPPLIER (Nombre del proveedor)
* ITEM CODE (Código del producto)
* ITEM DESCRIPTION (Descripción del producto)
* ITEM TYPE (Tipo de producto: licores)
* RETAIL SALES (Producto vendido en los dispensarios en dólares)
* RETAIL TRANSFERENS (Producto transferido a dispensarios en dólares)
* WAREHOUSE SALES (Producto vendido a licenciatarios de MC)

La base de datos cuenta con alrededor de 300,000 registros, que son suficientes para poder hacer particiones y validaciones.

A su vez con esta información podemos procesarlas para encontrar patrones en las tendencias de compras, hacer predicciones de ventas minoritas (Retail Sales), incluso predicciones de ventas por proveedor (Supplier).

[**Warehouse and Retail Sales.**](https://data.montgomerycountymd.gov/Community-Recreation/Warehouse-and-Retail-Sales/v76h-r7br/about_data)

V. IMPLEMENTACIÓN DEL MODELO

Por cuestiones dinamismo, es importante escoger un modelo entre regresión lineal y logístico, por ende, vamos a escoger el modelo que ocupe variables continuas (todo aquello que sea un valor numérico real), ya que un modelo logístico ocupa variables categóricas (todo aquello que entre un rango de cero a uno), esto conlleva modificar la tabla de la dataset.

Un modelo de Machine Learning está compuesto de la siguiente estructura para su óptimo funcionamiento:

1. Extract Transfrom Load (ETL) -Dataset
   1. Conocer la dataset
   2. Limpia de datos
      1. Imputación
      2. Remover variable redudantes
   3. Definir variable a predecir con respecto a sus independientes.
   4. División de la muestra poblacional
      1. Set de entrenamiento 75%
      2. Set de validación 10%
      3. Set de prueba 15%
2. Algoritmo de machine learning
   1. Optimización de pesos
   2. Optimización de hiperparametros
3. Validación, Test y Entrenamiento
   1. Entrenamiento del modelo usando el algoritmo
      1. Primero entrenamiento
      2. Segunda validación
      3. Tercera prueba
   2. Evaluación del modelo
      1. Compara las pruebas y evaluarlas con estadística lineal
         1. R^2, MSE y RMSE

**V.1. Extracción Transformación y Cargar (ETL siglas en inglés)**

ETL se unos del paso más importante para la implementación de modelos lineales, ya que será el alimento para que nuestro modelo aprenda.

Ocuparemos las siguientes librerías para el procesamiento de datos.

* Pandas
* Numpy
* Matplotlib

Captura de pantalla de computadora

Descripción generada automáticamenteVersion1 paso a paso del procesamiento de datos: [GitHub](https://github.com/KOKAS-3o14TOS/-Machine_Learning-/blob/main/DATASET_RETAIL_SALES/VERSION1/RETAIL_SALES.ipynb)

Img 1. Visualización de la Dataset original.

Texto

Descripción generada automáticamenteVisualizamos a más profundidad los datos.

Img 2. Observación de tipos de variable de cada columna.

Captura de pantalla de computadora

Descripción generada automáticamente con confianza mediaEchamos un vistazo general de los datos de cómo están estructurados.

Img 3. Estructura de la dataset, mean, std, min, max, etc..

Al conocer un poco más la dataset, es importante replantarnos con exactitud cuántos valores hay con NAN, que son valores que aportan ruido a nuestro modelo.

Imagen que contiene Tabla

Descripción generada automáticamenteImg 4. Datos con Nan de la dataset.

Es importante eliminarlos para evitar ambigüedades, debidos a que representa aproximadamente 0.055% de la dataset total, no hay mucho impacto el eliminar estos los valores.

Cambiamos las variables categóricas por valor numérico para poder procesarlo con nuestro modelo. Ocuparemos codificación numérica. Lo que hacemos es cambias cada iteración por la cantidad de repetición que hay en la columna, aplicaremos esta columna para SUPPLIER, ITEM DESCRIPTION y ITEM TYPE.

Pantalla de computadora con letras

Descripción generada automáticamente con confianza media

Img 5. Visualización de cambio de SUPPLIER, ITEM DESCRIPTION y ITEM TYPE a valor numérico.

En este momento es sumamente importante encontrar correlaciones de las variables dependientes con la variable independiente.

Interfaz de usuario gráfica, Aplicación, Calendario

Descripción generada automáticamente con confianza mediaGraf 1. Gráfica de correlaciones entre variables.

A simple vista se ve una correlación importante de RETAIL TRANSFERERS con RETAIL SALES, no obstante, requerimos de una visualización más exacta de aquellos variables independentes que influyen en la variable RETAIL SALES.

Substraemos todos aquellos valores mayores a 0, esto significa que visualizaremos las correlaciones positivas con respecto a RETAIL SALES.

Texto

Descripción generada automáticamenteImg 6. Correlaciones positivas a negativas de las variables.

Vamos a visualizar con mayor detalle RETAIL TRANSFERS y WAREHOUSE SALES.

Gráfico, Gráfico de dispersión

Descripción generada automáticamente

Graf 2. Correlación de RETAIL SALES con RETAIL TRANSFERS.

Gráfico, Gráfico de dispersión

Descripción generada automáticamenteEl comportamiento de esta variable es lineal debido a su tendencia, podemos deducir que esta variable tiene un alto impacto positivo para poder describir el comportamiento de RETAIL SALES.

Graf 3. Correlación de RETAIL SALES con RETAIL TRANSFERS.

En la gráfica 3 existe cierto patrón, no obstante, debido a su dispersión de esto origina que disminuya la correlación con la variable de investigación.

Evaluamos las variables con estadística generar para una previsualización de nuestro modelo.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Img 7. Tabla de regresión lineal con indicadores de tendencias estadísticas.

Con siete features para el label retail sales, se llega a conseguir a aproximadamente con una r cuadrada de 96%, nuevamente sobre sale el coeficiente RETAIL TRANSFERS que cuenta con un 97% de impacto en el modelo.

Con estos resultados relativos, da pie a empezar a seccionar nuestra dataset en segmentos de entrenamiento, validación y prueba.

* Set de entrenamiento 75% → 230,607 datos
* Set de validación 10% → 30,747 datos
* Set de prueba 15% →46,121 datos

**V.2. Implementación de Algoritmo de Machine Learninng**

Como se ha mencionado anteriormente se ocupará regresión lineal para predecir las ventas al por menor.

Un modelo de Machine Learning se caracteriza por su auto ajuste de parámetros para mejor las predicciones o los resultados de un modelo, por lo que esta es la gran diferencia de Machine Learning a un modelo de regresión con simples estadísticas.

Estos algoritmos cuentan con el siguiente funcionamiento.

1. Parámetros de entrada, Bias, Features y Labels.
2. Escalamiento de los samples de entrenamiento, prueba y validación.
3. Algoritmo de autoajuste Descenso por gradiente que simula el autoaprendizaje.
4. Validación con los datos
5. Evaluar el modelo con el segmento de prueba.
6. Visualización del error.

Entonces, los features serán nuestras entradas para el modelo que nos permitirá predecir la etiqueta label que es nuestra variable independiente.

Features → Year, Month, Supplier, Item Description, Item Type, Retail Transfers y Warehouse Sales.

Label → Retail Sales.

Lo anterior estará encapsulado de la siguiente manera

Datos de entrenamiento 75%

* X\_train
* Y\_train

Datos de validación 10%

* X\_val
* Y\_val

Datos de prueba 15%

* X\_test
* Y\_test

Después de tener todo estructurado para alimentar el algoritmo, a los samples X\_train, X\_val y X\_test es necesario de estandarizarlos o normalizarlos, con el fin de aumentar la velocidad de convergencia y que las entradas estén dentro de un tipo de dominio en igual.

Existen muchas técnicas de normalización y estandarización, para cuestiones prácticas ocuparemos la técnica Mean-Max Scaling.

**Mean-Max Scaling**

Scaled → Array estandarizado de la columna

X\_ij → Iteración de la columna en valor

Mean\_i → Promedio de la columna

Texto

Descripción generada automáticamenteMax\_i → Máximo de la columna

Alg 1. Mean-Max Scaling.

Esta consiste en normalizar el los features para que estén dentro de un rango de -1 a 1, ya que al restar por la media los datos estarán alrededor del cero y al dividirlos se localizan entre un rango de -1 a 1 aproximamente.

Al tener nuestros samples de entrenamiento, validación y prueba, podemos empezar a entrenar nuestro modelo con los sample de entramiento.

El entrenamiento consiste auto-ajustar los coeficientes del modelo dependiendo del error de predicción de la muestra, matemáticamente lo podemos demostrar con la fórmula de descenso por gradiente, esta surge del principio de la [regla de la cadena](https://math.stackexchange.com/questions/70728/partial-derivative-in-gradient-descent-for-two-variables/1695446#1695446), para mayor información, cabe mencionar que es el principio básico de las regresiones logísticas y de los modelos de Deep learning.

**Gradient Descent**

→ ir o learning rate

→valores de iteraciones del dataset

→Hipótesis

→Valor del label en esta instancia de la hipótesis

→Valor de x en la hipótesis de esa iteración

→Valor del parámetro o theta dado por nosotros

La fórmula anterior consiste en una serie de pasos secuenciales repetitivos para alcanzar el mejor ajuste de los pesos, estos pasos son los siguientes:

1. Estructuración de la hipótesis (Nuestro modelo de regresión lineal)
2. Encapsulamos la función hipótesis para estructuración de la función de costo (Mean Squared Error)
3. Encapsulamiento de la funcion costo para la estructuración de la función descenso por gradiente
4. Texto

   Descripción generada automáticamenteActualización de los coeficientes

Alg 2. Gradient Descent.

**V.3. Validación, Test y Entrenamiento**

Por último, validación del algoritmo, que es medir el error de precisión de nuestro modelo entrenado, para ello la segmentación de las muestras de validación (test y validation).

Al tener muestras reales de la variable a predecir, alimentamos nuestro modelo entrenando para calcular y visualizar que tanto se equivoca nuestro modelo en predecir por el outlayer esperado de la muestra real y calculamos su error con el MSE o función de costo. Esto se puede graficar para observar el error.

**Mean Squared Error**

Mse → Mean Squared Error

y\_i → Iteración de la columna en valor

Texto

Descripción generada automáticamentehat-y\_i → Promedio

Alg 3. Mean Squared Error.

**VI. Algoritmo completo de una regresión lineal (Modelo 1)**

Prácticamente, su funcionamiento se vería de la siguiente manera.

Cabe recalcar que primeramente se evalúa la muestra de validación y luego el segmento prueba.

Diagrama

Descripción generada automáticamente

Diag 1. Funcionamiento del algoritmo de regresión lineal

En código completo se vería de la siguiente manera.

Texto

Descripción generada automáticamente

Alg 4. Escalamiento de los samples del algoritmo de machine learning de regresión lineal

Texto

Descripción generada automáticamente

Alg 5. Encapsulamiento del funcionamiento de Gradient Descent, Testing Model, MSE (Test y Val) y Plot.

Tuvimos que dividir el código completo, ya que en la sección del scaling se hace para cada uno de los samples de validez.

Gráfico, Gráfico de barras

Descripción generada automáticamente

Texto

Descripción generada automáticamenteUna vez ya planteado el funcionamiento básico de los algoritmos de machine learing, podemos evaluar nuestro modelo hecho-línea a línea, con finalidad de empezar a observar su funcionamiento para predecir la variable RETAIL SALES a futuro, además que tan rentable fue implementar un modelo de estos desde cero.

Img 8. Resultados numéricos del modelo 1.

Gráfico, Gráfico de barras

Descripción generada automáticamenteEs posible precisar que el modelo está muy mal optimizado debido a que se tardó entrenado 81 min con parámetros de 1000 épocas de entrenamiento con una tasa de aprendizaje del 0.5, y su rendimiento es prácticamente nulo, ya que el error MSE es de train(306), val(210) y test(269) unidades de variación para predecir las ventas por menor a futuro, además el modelo en promedio tiene una capacidad descriptiva del 48%.

Graf 4. MSE Modelo 1 contra variables.

En general el modelo cuenta con error MSE muy alto, esto conlleva que sus estimaciones varíen mucho y sean imprecisas.

Graf 5. R^2 del Modelo 1 contra variables.

Gráfico

Descripción generada automáticamenteDebido a su imprecisión el modelo se llega a considerar underfiting, esto se traduce a un modelo muy malo para predecir tendencias, y que no pude predecir ni las métricas con las que entreno.

Graf 6. Predicciones Train vs Real Modelo 1.

El modelo cuenta con underfitting por su rendimiento precario, pues en el modelo no pudo predecir los valores con los que se entrenó, esta es la prueba más sencilla.

Gráfico

Descripción generada automáticamente con confianza media

Graf 7. Predicciones Validation vs Real Modelo 1.

Gráfico

Descripción generada automáticamente

Gráfico, Gráfico de barras

Descripción generada automáticamenteGraf 8. Predicciones Test vs Real Modelo 1.

El desempeño de nuestro algoritmo fue muy malo, esto se puede deber a múltiples factores, no obstante, los que destacan son: Optimización, Scaling, y Tiempo de entrenamiento.

**Optimización:** Nuestro modelo, utiliza estructuras cíclicas, esto quiere decir que los procesos son loops que van calculando por iteraciones, al utiliza tipos de estructuras así la complejidad sube a O(n), entre mayor la dataset, menos eficiente se vuelve y tarda más.

**Scaling:** Hay distintos tipos de escalamiento, y cada uno tiene la posibilidad de que se adapte distinto a tu dataset favoreciendo al mejoramiento del entrenamiento, los más comunes son Mean-Max y Min-Max, por los que, es importante probar con diferentes escalamientos.

**Tiempo de entrenamiento:** Este punto tiene una correlación con la optimización del código, ya que, si entrenamos el modelo con una cantidad alta de épocas, va a tardar 10^n iteraciones debido a su complejidad del algoritmo.

**VII. Algoritmo con Framework de una regresión lineal (Modelo 2)**

De otro modo, es posible mejorar drásticamente la predicción de nuestro modelo debido a que ya existen librería optimizada para el uso de algoritmos de machine learning, estas librerías constan de codigo encapsulado como los es nuestro algoritmo hecho por su servidor anteriormente, la diferencia radica en que estos están optimizados.

* Scikit-learn
* TensorFlow
* PyTorch
* Keras

Por ende, vamos a implementar el mismo algoritmo con framework para comparar el primer modelo con este, y ver cuánto mejora. Código: [GitHub](https://github.com/KOKAS-3o14TOS/-Machine_Learning-/blob/main/REGRESION_LINEAL/REGRESION_LINEAL_FRAMEWORK.py)

Utilizaremos el Framework (Scikit-learn). El modelo nos arroja la siguiente información:

Texto

Descripción generada automáticamente

Img 9. Resultados numéricos del modelo 2.

El promedio MSE es 26 unidades, esto significa que, nuestro modelo tendrá un error de predicción de 26 unidades. Asimismo, el promedio de R^2 es de 0.94, que representa que nuestro modelo describe el comportamiento con una precisión del 94%. Se tardó entrenándose 0.03 segundos.

Graf 9. MSE del Modelo 2 contra variables.

Gráfico, Gráfico de barras

Descripción generada automáticamenteGráfico

Descripción generada automáticamenteComo podemos ver nuestro modelo no está sobre ajustado, ya que el MSE de Train es de 27 unidades, sino seria de cero, en cambio el error MSE para test es de 25 y val de 24 unidades, por lo que concretamos que nuestro modelo predice los RETAIL SALES a futuro años con margen de error mucho más pequeño comparado al modelo 1.

Graf 10. R^2 del Modelo 2 contra variables.

En teoría, el modelo cuenta con coeficientes con fuerza explicativa para predecir RETAIL SALES a futuro con un 94% de efectividad aproximadamente de todas las pruebas.

Gráfico

Descripción generada automáticamente

Graf 11. Predicciones Train vs Real Modelo 2.

El modelo no tiene ovedrfitting, ya que su error MSE de 27, y tampoco es underfitting debido a su precisión descriptiva. No obstante, es la primera prueba.

Graf 12. Predicciones Validation vs Real Modelo2

En esta prueba el modelo obtuvo un MSE menor a los otros dos, y con una R^2 de 95% de poder descriptivo.

Gráfico

Descripción generada automáticamente

Graf 13. Predicciones Test vs Real Modelo 2.

En esta prueba los rendimientos del modelo fueron de MSE de 25 unidades, su poder descriptivo del 94%, podemos decir que este modelo está fitting con una eficiencia del 94% debido a que supero las tres pruebas con valores decentes.

**VIII. Algoritmo RandomForest Modelo 3**

Vamos ahora a comparar el modelo de regresión lineal con otro modelo más robusto, y visualizar si hay mejoras.

Ocuparemos el algoritmo RandomForest, es un método que consta de una arquitectura bagging con arboles de desción, este consta de tener multples arboles pequeños que se entrenar en paralelo.

Esto se traduce, que tendremos pequeños arboles de deciones que van subdiviendo las deciones hasta estar en el rango esperado.

No entramos a detalle del modelo matemático, solo de su implementación.

Librerias a ocupar de machien learning son:

Texto

Descripción generada automáticamenteSklearn y Skopt; esta última librería la ocuparemos para optimizar los hiper-prametros de nuestro modelo con el fin obtener mejores estimaciones.

Código: [Github](https://github.com/KOKAS-3o14TOS/-Machine_Learning-/blob/main/REGRESION_LINEAL/RANDOM_FOREST_REGRESION_LINEAL.py)

En la estimación de los hiperparametros en arboles, hay miles de técnicas, la que ocuparemos se llama [Bayesian Optimization](https://python.plainenglish.io/master-the-power-of-scikit-optimize-a-step-by-step-guide-4346d3a484ea), consuiste en la búsqueda del conjunto óptimo del hiperparamentro modelando la función de rendimiento como una función probabilística, y busca en las áreas más relevantes, la gran vnetja de esta técnica es que explora de manera inteligente el espacio del hiperparametro.

Diagrama

Descripción generada automáticamente

Diag 2. Funcionamiento de RandomForest con Bayesian.

Al entender funcionamiento del modelo, es posible correr el algoritmo.

Img 10. Resultados numéricos del modelo 3.

El modelo obtuvo una precisión del 95%, mayor a modelo 2, este cuenta con una precisión del 94%, un porciento mejor.

El modelo le tomó 53 minutos en optimizarse y entrenar.

Gráfico, Gráfico de barras

Descripción generada automáticamente

Graf 14. MSE del Modelo 3 contra variables.

Parece que nuestro modelo tuvo un sobre ajuste, ya que disminuyo la prueba train, y las otras predicciones tuvieron 3 unidades más de error.

Esto quiere decir que tenemos un overfitting de 3 unidades.

Gráfico, Gráfico de barras

Descripción generada automáticamente

Graf 15. R^2 del Modelo 3 contra variables.

El modelo de RandomForest tiene un poder explicativo de las tendencias de ventas minorista del 95% con un margen de error del 5%.

Gráfico

Descripción generada automáticamente

Graf 16. Predicciones Train vs Real Modelo 3.

Gráfico

Descripción generada automáticamenteComo podemos observar, el modelo tiene un sobre ajuste, que impide que a la hora de predecir no sobre pase el valor y esto es muy notorio en la gráfica.

Graf 17. Predicciones Val vs Real Modelo 3.

El modelo cuenta con un sobre ajuste, pero mejoró en un 1% de predicción, y de MSE se redujo también una unidad.

Gráfico

Descripción generada automáticamente

Graf 18. Predicciones Test vs Real Modelo 3.

El modelo fue un poco más certero, está overfitting.

**IX. CONCLUSIONES**

Los últimos dos modelos de framework dieron rendimientos óptimos para predecir a grasso modo las ventas al por menor de las bebidas alcohólicas de las tiendas de EE. UU en la zona de Montgomery.

Parte del éxito se debió a la dataset, esta cuenta con 300,000 datos para alimentar los algoritmos, esto nos da entender que la esencia de los algoritmos radica en conocer los datos para saber interpretar patrones, por lo que podemos decir que el talón de Aquiles de esta tecnología es la información, por ende es sumamente importante contar con dataset estructuradas para evitar meterle mano a los datos y ensuciarlos, lo bueno es que se puede ocupar herramientas como lo es excel para crear tus bases de datos sin ocupar herramientas de la nube de pago, a su vez estas herramientas son versátiles para visualizar y predecir tendencias, parte de estos modelos dieren a luz desde los 80°, hoy en día basta con solo correr tres líneas de código para que el algoritmo tenga vida.

Nuestra hipótesis dice:

*Podría ser una respuesta el implementar modelos de Machine Learning para impulsar los negocios minoritas, tomando en cuenta su complejidad, su asertividad y su implementación.*

Los modelos de machine learning dieron buenos resultados al momento de predecir las ventas minorita a futuros, el algoritmo de Framework de regresión mostro rendimientos sorprendentes, el modelo cuenta con un 94% de asertividad para predecir y se tardó en entrenar 0.03 segundos. Un excelente modelo para ampliar el panorama de las ventas futuras., sin embargo, el primer modelo nos ayudo a visualizar con mayor claridad como funciona estos algoritmos, y por último el algoritmo RandomForest, tuvo un desempeño impecable.

Para mejorar la implementación de los modelos es necesario hacer imputación de los datos y eliminar valores atípicos. La implementación fue sencilla debido a que tengo experiencia programando, pero solo un porcentaje chico cuenta con estas habilidades, esto significa que parte de la población mexicana se le complica implementar estos modelos, y esto me lleva abordar el siguiente punto crítico.

En México en general existe una brecha de conocimiento tecnológico en múltiples sectores, parte de la esencia o capacidades para poder trabajar con estas herramientas se focalizan en ganas de aprender nuevas tecnologías y saber programar.

Esto a su vez nos lleva a indagar que se traduce a dinero el conocimiento, tiene precio y el tiempo también se traduce en dinero. Aprender estos temas me llevó 5 semana full, a que añadirle que aprendí eso conocimientos en la mejor Universidad del país, y como alumno promedio pago una colegiatura alrededor de 151,272pesos, el ciclo escolar dura 18 semanas, por cada semana estoy pagando 8,404 pesos, y de acuerdo con la zona norte del país una persona en promedio gana 1,562 pesos a la semana (Dainzú Patiño, 2022).

Por los conocimientos adquiridos de machine learning estoy pagando 5.3 veces el salario mínimo en una semana. En una conclusión genérica, en México enfocarse a las áreas de la tecnología es muy caro, por lo que no es rentable invertir en estas áreas para las pequeñas empresas minoritas, gran parte de estas viven del día al día.

REFERENCIAS

*Artificial Intelligence: A Modern Approach, 4th Global ed.* (2022). Berkeley.edu. <https://aima.cs.berkeley.edu/global-index.html>

Elizabeth Meza Rodríguez. (2022, August 18). *3 de cada 4 tienditas desaparecen cuando llega una tienda de conveniencia a la colonia*. El Economista; El Economista. <https://www.eleconomista.com.mx/el-empresario/3-de-cada-4-tienditas-desaparecen-cuando-llega-una-tienda-de-conveniencia-a-la-colonia--20211105-0053.html>

Ruiz-Healy, E. (2024, February 22). *El impresionante crecimiento del sector minorista mexicano*. El Economista; El Economista. <https://www.eleconomista.com.mx/opinion/El-impresionante-crecimiento-del-sector-minorista-mexicano-20240221-0113.html>

Urrego, N. (2023, July 10). *Codificación de variables categóricas con Python y R: Técnicas y conceptos clave.* Medium; Medium. <https://nicolasurrego.medium.com/codificaci%C3%B3n-de-variables-categ%C3%B3ricas-t%C3%A9cnicas-y-conceptos-clave-bca17b6164b2#:~:text=La%20codificaci%C3%B3n%20de%20frecuencia%20es,cada%20categor%C3%ADa%20en%20los%20datos>.

‌

¿A cuánto asciende el salario mínimo en México y para qué debe alcanzarte?

By Dainzú Patiño Container: Expansión Year: 2022 URL: <https://expansion.mx/economia/2022/07/19/salario-minimo-en-mexico-semana-quincena-mes-ano>

‌

‌

‌